

Year One Report

AutoHelp

**An 'Intelligent' Case-Based Help Desk Providing
Web-Based Support for EOSDIS Customers**

A Concept and Proof-of-Concept Implementation

**Attention: Karen L. Moe, Walt Truszkowski
Code 520**

**Christine M. Mitchell, Principal Investigator
David A. Thurman, Co-Principal Investigator
Center for Human-Machine Systems Research
School of Industrial & Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332-0205**

**(404) 894-4321 (Mitchell)
(404) 894-2318 (Thurman)
(404) 894-2301 (fax)
[cm,dave]@chmsr.gatech.edu**

January 1999 .?

GT: E-24-T12

NASA GSFC: NAG 5 3356

Summary for Year One

AutoHelp is a case-based, Web-accessible help desk for users of the EOSDIS. It uses a combination of advanced computer and Web technologies, knowledge-based systems tools, and cognitive engineering to offload the current, person-intensive, help desk facilities at the DAACs.

As a case-based system, AutoHelp starts with an organized database of previous help requests (questions and answers) indexed by a hierarchical category structure that facilitates recognition by persons seeking assistance. As an initial proof-of-concept demonstration, a month of email help requests to the Goddard DAAC were analyzed and partially organized into help request cases. These cases were then categorized to create a preliminary case indexing system, or category structure. This category structure allows potential users to identify or recognize categories of questions, responses, and sample cases similar to their needs.

Year one of this research project focused on the development of a technology demonstration. User assistance 'cases' are stored in an Oracle database in a combination of tables linking prototypical questions with responses and detailed examples from the email help requests analyzed to date. When a potential user accesses the AutoHelp system, a Web server provides a Java applet that displays the category structure of the help case base organized by the needs of previous users. When the user identifies or requests a particular type of assistance, the applet uses Java database connectivity (JDBC) software to access the database and extract the relevant cases.

A presentation on January 29, 1998 included an on-line demonstration of the current AutoHelp system, including examples demonstrating how a user might request assistance via the Web interface and how the AutoHelp case base responds.

The presentation described the preliminary structure of AutoHelp including DAAC data collection, case definition, and organization to date, as well as the AutoHelp architecture. It concluded with the year 2 proposal to more fully develop the case base, the user

interface (including the category structure), interface with the current DAAC Help System, the development of tools to add new cases, and user testing and evaluation at (perhaps) the Goddard DAAC.

In addition to the proof-of-concept demonstration and an on-line demonstration, the AutoHelp project was described in a paper presented at the 1997 IEEE Conference on Systems, Man, and Cybernetics (Appendix A) and with several PowerPoint presentations that are available both in paper form and electronically (Appendix B).

Appendix A

Thurman, D. A., Tracy, J. S., & Mitchell, C. M. (1997). Design of an 'intelligent' Web-based help desk system. *Proceedings of the 1997 IEEE International Conference on Systems, Man, and Cybernetics*, Orlando, FL, 2198-2203.

Design of an 'Intelligent' Web-Based Help Desk System

David A. Thurman, Jeffrey S. Tracy, and Christine M. Mitchell
Center for Human-Machine Systems Research
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205

ABSTRACT

This paper describes a project that extends the concept of help desk automation by offering World Wide Web access to a case-based help desk. It explores the use of case-based reasoning and cognitive engineering models to create an 'intelligent' help desk system, one that *learns*. It discusses the AutoHelp architecture for such a help desk and summarizes the technologies used to create a help desk for NASA data users.

1. INTRODUCTION

Many organizations, particularly in computer hardware and software areas, provide extensive customer support via telephone 'help desks.' This assistance depends primarily on human expertise to respond to user questions. In many cases, the help desk staff must answer the same questions repeatedly. Such support is expensive, requiring a large technical staff. Due to budget limitations, many organizations are trying to serve an increasing user population while either maintaining or reducing the size of their help desk staff. This project demonstrates the use of emerging automation technologies to potentially permit better utilization of increasingly scarce human resources.

Case-based reasoning (CBR) is one such technology whose potential for widespread application is receiving a great deal of attention (Rapoza, 1996). Case-based reasoning stores its intelligence as a set of experiences or 'cases'. It is based on the theory that decision makers recall past experiences to formulate solutions to new problems (Riesbeck & Schank, 1989). Similarly, a case-based reasoning system stores memorable problems and their solutions in its knowledge base and retrieves them when it encounters a similar problem. A case-based system is appealing as the underlying technology to support a help desk, since many requests for help are reoccurring or variations of previous requests (Allen, 1994; Kriegsmann & Barletta, 1993).

Currently, most case-based reasoning systems for help desk applications are designed to provide in-house staff support. Typically, these tools are used by help desk staff to personally respond to requests for assistance—a predominantly manual system with the case base providing decision support for help desk staff (Allen, 1994; Fine, 1995; Kriegsmann & Barletta, 1993; Pickering, 1993; Wallace, 1994). More recently, however, attention has focused on designing tools that allow users requiring assistance to access the case base themselves—one potential step towards help desk automation (Pickering, 1993).

This paper describes a project that extends the concept of help desk automation by offering World Wide Web access to a case-based help desk. The goal is to provide users direct access to the knowledge in the case base and enable the case-based reasoner to respond to requests for assistance. Help desk staff intervention is necessary only when help desk automation cannot provide a suitable response.

Technologically, a Web-based help desk appears feasible. One goal of this project, however, is to integrate and demonstrate a prototype of the required hardware and software components for an end-to-end system. Another goal is to explore the use of cognitive engineering models to define both the required case structure and the access strategies for an 'intelligent' help desk system. This system is called 'intelligent' because it *learns*. For requests to which the help system cannot respond, it provides a medium through which help desk staff can concurrently answer a user request *and* create and index a new case for the help desk database. To illustrate and assess the viability of this 'intelligent' system, it is being implemented as a proof-of-concept prototype for a NASA help desk system—the user support organization of EOSDIS.

2. EOSDIS

As a component of NASA's Mission to Planet Earth and the larger U.S. Global Change Research program, the Earth Observing System Data and Information System (EOSDIS) manages data from NASA Earth science satellites and field measurement programs. EOSDIS provides data archiving, distribution, and information management services to a wide range of users including international scientists, educators, and students. EOSDIS stores data collected from science satellites and experiments, develops and distributes tools for data manipulation, and generates standard data products known as 'data sets.'

EOSDIS is a distributed architecture in which Earth science data is stored and made available to users via the nine Distributed Active Archive Centers (DAAC) shown in Figure 1. Each DAAC stores data sets and tools corresponding to a particular type of science data (e.g., the Jet Propulsion Laboratory DAAC provides physical oceanography data sets and tools). Each DAAC also houses a user support organization, i.e., help desk staff, which assists users in data acquisition, search, access, and usage. Typically, the user support organization has a single person who serves as the user support coordinator and several data support teams, composed of individuals having specialized knowledge about particular data sets (e.g., ocean temperatures or precipitation measures). The user support coordinator is the initial point of contact for

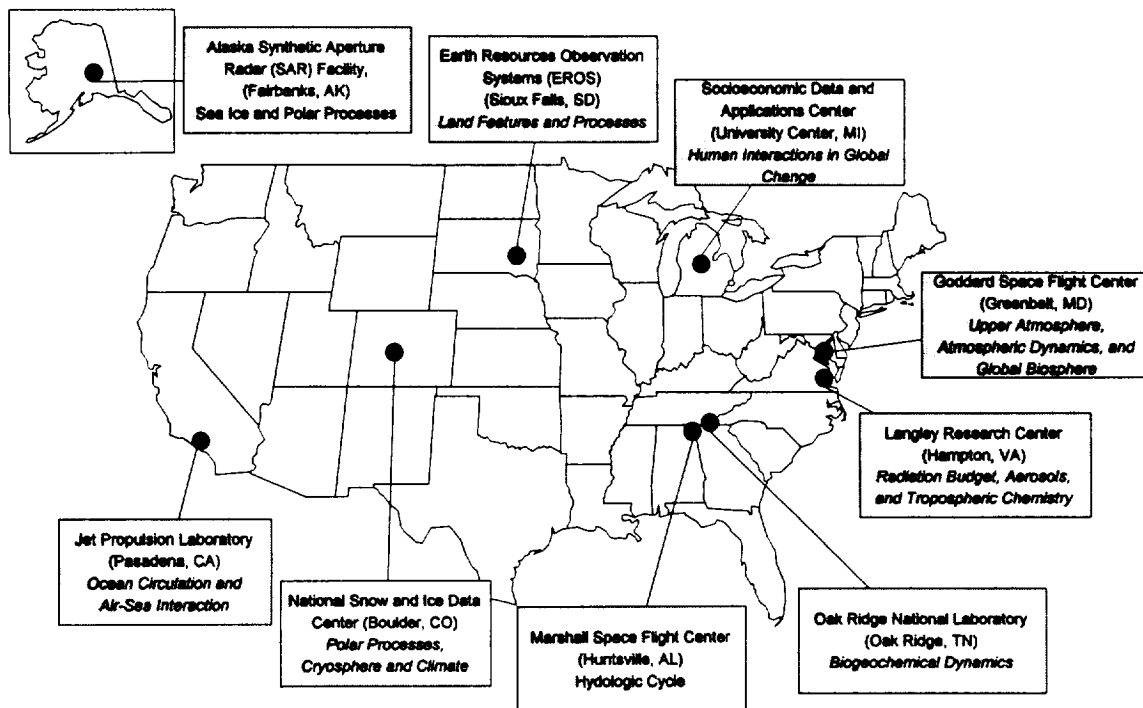


Fig. 1. EODIS Distributed Active Archive Centers (DAACs)

requests that arrive via electronic and postal mail, facsimile, and telephone. When a request arrives, the support coordinator either responds, forwards it to the appropriate data support team, or forwards it to another DAAC (Figure 2).

This type of assistance is labor-intensive and as a result, very costly. Due to budget constraints, NASA is attempting to reduce the operations costs associated with EODIS. However, due to outreach efforts by the EOS project, the number of users served by EODIS is growing exponentially and is expected to exceed one million users by the turn of the century. Thus, maintaining the current mode of help desk operations will be very difficult. A Web-based 'intelligent' help desk might reduce costs while meeting expanding user demand. The goal of this project is to demonstrate the feasibility of an 'intelligent' help desk that meets this increased demand and potentially lowers the cost of providing help desk support.

3. METHOD

The design and development of a Web-based help desk requires tools for modeling and representing the knowledge required to support help desk operations and technologies for making this knowledge available to potential users. Several conceptual tools may be useful in the design and implementation of a Web-based help desk. These include case-based reasoning, cognitive engineering, and the design of 'intelligent' systems. Available technologies which have a potential role in the Web-based help desk include the World Wide Web, the Java programming language, and commercial case-based reasoning systems and database management systems.

Case-Based Reasoning. Recognition-primed decision making is an emerging theory of human problem solving and decision making research (Zsombok & Klein, 1997). An alternative to classical behavioral decision theory (Hogarth, 1987), recognition-primed decision making is based on widespread field studies of expert decision makers and posits that expert decision makers do not generate a space of potential solutions, rank them via a set of criteria, and select the best. Rather, experts 'recognize' situations, recall past solutions to similar problems, and adapt them to solve new problems.

Case-based reasoning can be thought of as a computational implementation of recognition-primed decision making. A case is a contextualized piece of knowledge that teaches a lesson fundamental to achieving the decision maker's goals (Kolodner, 1993). It contains information on how a task was carried out, how a piece of knowledge was applied, or what strategies for accomplishing a goal were used (Kolodner, 1993). There are two parts of a case-based reasoner that allow it to understand a problem based on past experience: *recalling* an old experience and *interpreting* the new situation in terms of the recalled experiences. The first is called the indexing problem: recalling the situation(s) most similar to a new situation. The second, interpretation, includes comparison of the new situation with previous situations, and if necessary, adapting the old experience to fit the new situation. A case-based reasoning system *learns* in two ways: through the accumulation of new cases and through the modification and extension of indices.

The application of case-based reasoning to real world problems has demonstrated that the knowledge engineering required to define the initial set of cases and specify the indices can be a major bottleneck and ultimately determine the success or

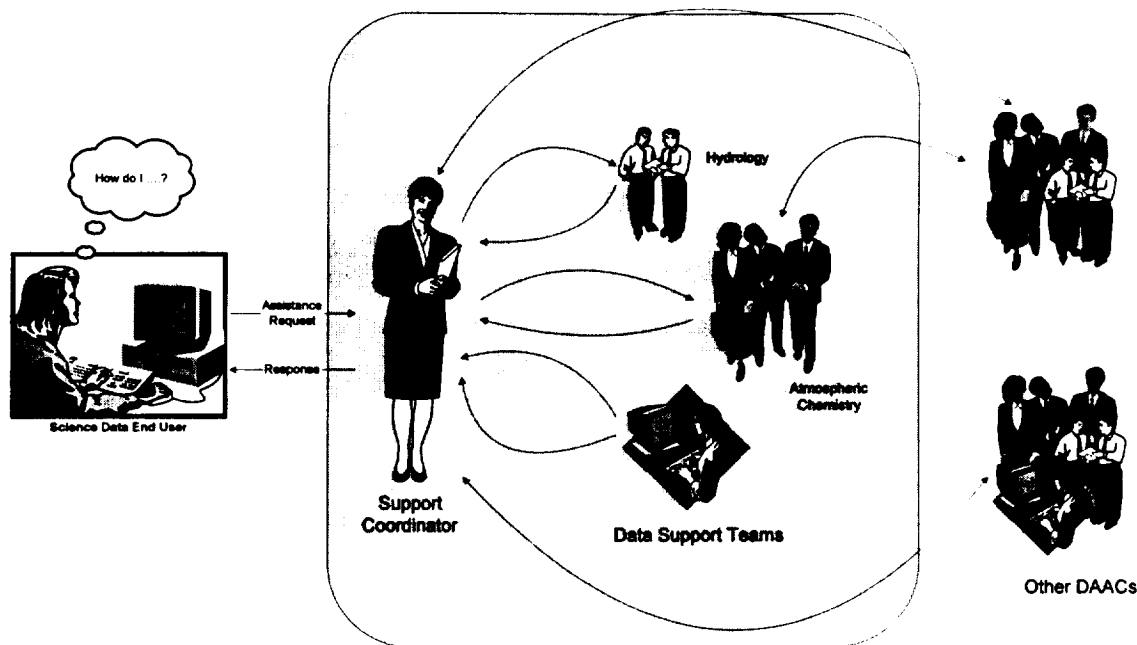


Fig. 2. Current DAAC help desk environment.

failure of the application (Wallace, 1994). A case-based reasoning system for a help desk begins with defining the structure and content of a case. Next, indexing structure and the means for adding new cases must be addressed. This research explores the use of cognitive engineering models to support the initial design and long-term maintenance of a case-based help desk.

Cognitive Engineering Models. Typically concerned with cognition in real world tasks, cognitive engineering uses models to design computer-based interfaces, aids, and training systems. Cognitive engineering models, such as the operator function model (Mitchell, 1996) and Rasmussen's models (Rasmussen, Pejtersen, & Goodstein, 1994), have been successfully used to design operator displays, aids, and tutoring systems (Mitchell, 1996; Vicente, Christofferson, & Perekliita, 1995). Empirical evidence shows that artifacts (e.g., displays) designed using models such as the operator function model enhance user performance when compared to conventionally designed artifacts (e.g., Mitchell, 1996; Thurman & Mitchell, 1995).

This research explores the use of cognitive models to define both the structure and content of the help desk case base and the interfaces for both users who seek assistance and the help desk staff who maintain and enhance the computer-based help system. A design based on models of help desk functions and assistance requests may help users 'recognize' their request in the knowledge base and identify possible solutions.

'Intelligent' Systems. One goal of this project is the design of an 'intelligent' system. In this situation, an 'intelligent' system is defined as one which does not make the same mistake again, i.e., it *learns*. From both artificial intelligence and user interface perspectives, the issue of designing a case-based system that learns over time is challenging. This project explores the design of hybrid

intelligence that acquires information automatically. Whenever the case-based help desk requires human intervention to address a novel assistance request, the interface for the help desk staff allows the staff to concurrently answer a request and update the case-based reasoning system. The latter requires software that automatically creates a new case and new indices, and allows the help desk staff to inspect, repair, and validate the new addition to ensure that it is correct and consistent with other cases and indices.

Available Technologies

Available commercial off-the-shelf technologies provide the necessary capabilities to design a Web-based CBR help desk. These technologies include the World Wide Web, the Java programming language, case-based reasoning systems, and relational and object-oriented databases.

World Wide Web. The recent explosion and increasing availability of World Wide Web technologies for distribution of information to a varied and widespread user community makes it an attractive medium through which to provide help to geographically distributed users. World Wide Web browsers are now commonplace (i.e., on most desktop computers) and electronic documents can be easily transformed into Web pages via the HyperText Markup Language (HTML). The widespread use of the Web, combined with the ease with which a World Wide Web server can be placed on the Internet, makes this an attractive set of technologies to consider for this research project.

HTML, however, displays predominantly static information. Although Common Gateway Interface (CGI) and HTML extensions can provide fill-in request forms and retrieve information from databases, such applications run on the host computer, thus placing a heavy load on the server if there are multiple users. Given the projected one million EOSDIS users,

providing all user assistance by means of a common server is neither desirable nor feasible.

Java. Java™ runs on most platforms and with most Web browsers. Java applets are downloaded from a Web server and run on the user's computer. Thus, part of the processing load is shifted from the Web server to the user's computer. Moreover, applets can communicate with both the Web server and other servers such as DAAC databases or a case-based help server. The ability to create platform-independent, distributed applications with Java makes it a critical component for the Web-based help desk architecture.

Case-Based Reasoning Systems. Several commercially available case-based reasoning tools were evaluated. They include Inference Corporation's CBR Express and CasePoint, Cognitive Systems' REMIND, and Esteem Systems' ESTEEM (Allen, 1994). The comparative recency and somewhat unsophisticated nature (Allen, 1994) of the commercial CBR tools are a limitation. Some systems are still implemented in Lisp, others store cases in flat files rather than a database, and few, if any, provide a distributed client-server architecture that can operate over the Internet.

Databases. Existing relational databases such as Oracle™ and Sybase™ as well as object-oriented databases such as Illustra™ offer commercially available and widely used information storage, management, and analysis tools. These are viable alternatives to the case storage feature of commercial CBR systems.

4. AUTOHELP ARCHITECTURE

Based on the analysis of available tools and technologies, an AutoHelp architecture for an 'intelligent' help desk is proposed (see Figure 3). The AutoHelp architecture consists of a Web

server which provides access to HTML pages, Java applets containing the user interface and case structure information, and a case-based reasoning server which stores previous assistance requests and associated solutions. The case-based reasoning system is distributed across Java applets that contain the case structure information, and an Oracle database that contains the individual cases that make up the case base.

The user interface is composed of Java applets and uses concepts of recognition-primed decision making to assist the user in navigating through the case base. The support staff interface is also implemented in Java as a stand-alone application that concurrently allows help desk staff to respond to a user request and updates the case base.

A technology demonstration was conducted to ensure that this architecture has the necessary capabilities to support a distributed case-based reasoning help desk application for EOSDIS end users.

5. APPLICATION TO EOSDIS

The design of a Web-based intelligent help desk for EOSDIS consisted of two phases: cognitive engineering and the development of a proof-of-concept demonstration.

Cognitive Engineering Analysis

A field study of EOSDIS help desk operations was conducted, including visits to the various DAACs, interviews with help desk staff, and collection of sample help requests. Because the Goddard DAAC keeps extensive current and archival electronic records of each help request it receives, the process through which the request was answered, and its final disposition, it is a rich source of information on the types of assistance provided to EOSDIS users. Approximately 1500 email messages received and sent during 1996 were analyzed to build an initial

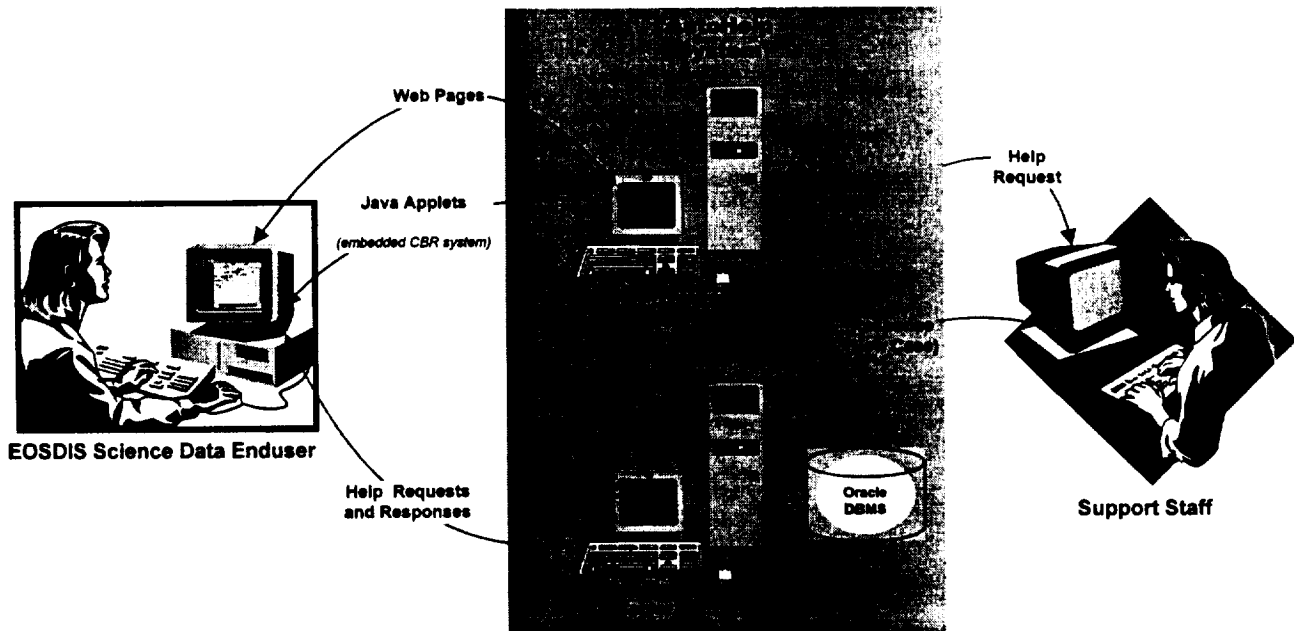


Fig. 3. AutoHelp Architecture

case structure.

Based on this analysis, a case was defined as all correspondence pertaining to a particular question asked in a help request. For instance, if a user requests information on ocean temperature, the help desk staff would respond with a listing of data sets containing ocean temperature data. These messages together would form one case in the case base.

Using the concepts of recognition-primed decision making, a model of user help requests was developed to organize the cases. They are organized in such a way that users can swiftly recognize the problem and associated assistance. Types of assistance requests were abstracted to multiple levels to facilitate indexing for the case base.

A sample of approximately 400 of the messages was examined more thoroughly to refine the case and index structures. Categories of requests containing few messages were merged with other categories to reduce complexity. A portion of the resulting indexing structure is shown in Figure 4. In addition, a model to guide the design of the interface for the help desk staff was created. The model, in conjunction with knowledge gained from the field study, helped specify the categories of activities the staff currently performs, and enumerate the capabilities of an AutoHelp implementation for EOSDIS.

Proof-of-Concept Demonstration. An AutoHelp implementation for EOSDIS was developed based on the user model and case indexing structure produced in the field study. This implementation consists of a Web server, Java applets, and a CBR server consisting of an Oracle database and WebLogic™ JdbcT3 software to provide applet to database connectivity (see Figure 5). This architecture has been successfully demonstrated to provide Web-based access to the case base of available EOSDIS help information.

Evaluation. AutoHelp will be evaluated in two phases. In the first phase, the capability of the AutoHelp implementation will be evaluated using requests received during May 1997. Evaluation criteria include the percentage of incoming help requests to which the AutoHelp implementation can provide satisfactory solutions and the support staff's ability to update both the user interface and the case base. In the second phase, the full prototype will be evaluated on site at the Goddard DAAC. Users will attempt to find the solutions to their questions via the user interface. Empirical data will be collected and analyzed to determine the effectiveness of the AutoHelp architecture.

6. CONCLUSION

Based on the analysis of available technologies, a Web-based 'intelligent' help desk is a viable alternative to current help desks. Case-based reasoning technology combined with the concepts of recognition-primed decision making can be useful in such a system. Experiences encoded as cases allow users to identify solutions to their own problems, reducing the burden on help desk staff. AutoHelp could provide a means to solve the problem of increasing user demand for assistance while reducing associated costs.

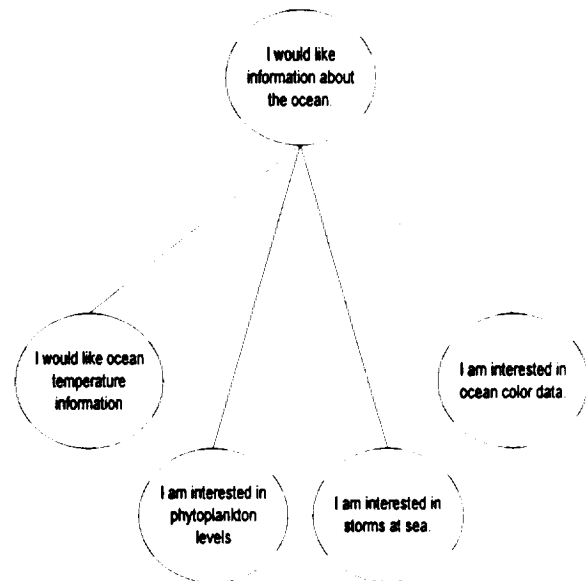


Fig. 4. Part of the case base index structure.

7. ACKNOWLEDGEMENTS

This research is supported in part by a NASA Goddard Space Flight Center Grant (Walt Truszowski, Technical Monitor). Product names are registered trademarks of their respective companies.

8. REFERENCES

- Allen, B. P. (1994). Case-based reasoning: Business Applications. *Communications of the ACM*, 37(3), 40-42.
- Fine, D. (1995, October 23). Help desk's new hat. *INFOWORLD*, 20, 67-71.
- Hogarth, R. M. (1987). *The psychology of judgement and choice*. (2nd ed. ed.). San Francisco, CA: Jossey-Bass.
- Kolodner, J. (1993). *Case-Based Reasoning*. San Mateo, CA: Morgan Kaufmann.
- Kriegsman, M., & Barletta, R. (1993). Building a case-based help desk application. *IEEE Expert*, 18-26.
- Mitchell, C. M. (1996). GT-MSOCC: Operator models, model-based displays, and intelligent aiding. In W. B. Rouse (Ed.), *Human/technology interaction in complex systems* (Vol. 8, pp. 67-172). Greenwich, CT: JAI Press Inc.
- Pickering. (1993, December 1). Is smart software the answer for help desks? *DATAMATION*, 42-46.
- Rapoza, J. (1996, September 30). A smart way to put help on the Web. *PC Week*, 13, 93.
- Rasmussen, J., Pejtersen, A. M., & Goodstein, L. P. (1994). *Cognitive systems engineering*. New York: John Wiley.
- Riesbeck, C. K., & Schank, R. C. (1989). *Inside Case-Based Reasoning*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Thurman, D. A., & Mitchell, C. M. (1995). A design methodology for operator displays of highly automated supervisory control systems. *Proceedings of the 6th IFAC/IFIP/IFOR/SEA Symposium on Analysis, Design, and Evaluation of Man Machine Systems*, Boston, MA.
- Vicente, K. J., Christofferson, K., & Pereklika, A. (1995). Supporting operator problem solving through ecological

interface design. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-25(4), 529-545.

Wallace, P. (1994, January 17). Help desk software. *INFOWORLD*, 55-58.

Zsombok, C. E., & Klein, G. (1997). *Naturalistic Decision Making*. Mahwah, NJ: Lawrence Erlbaum Associates.

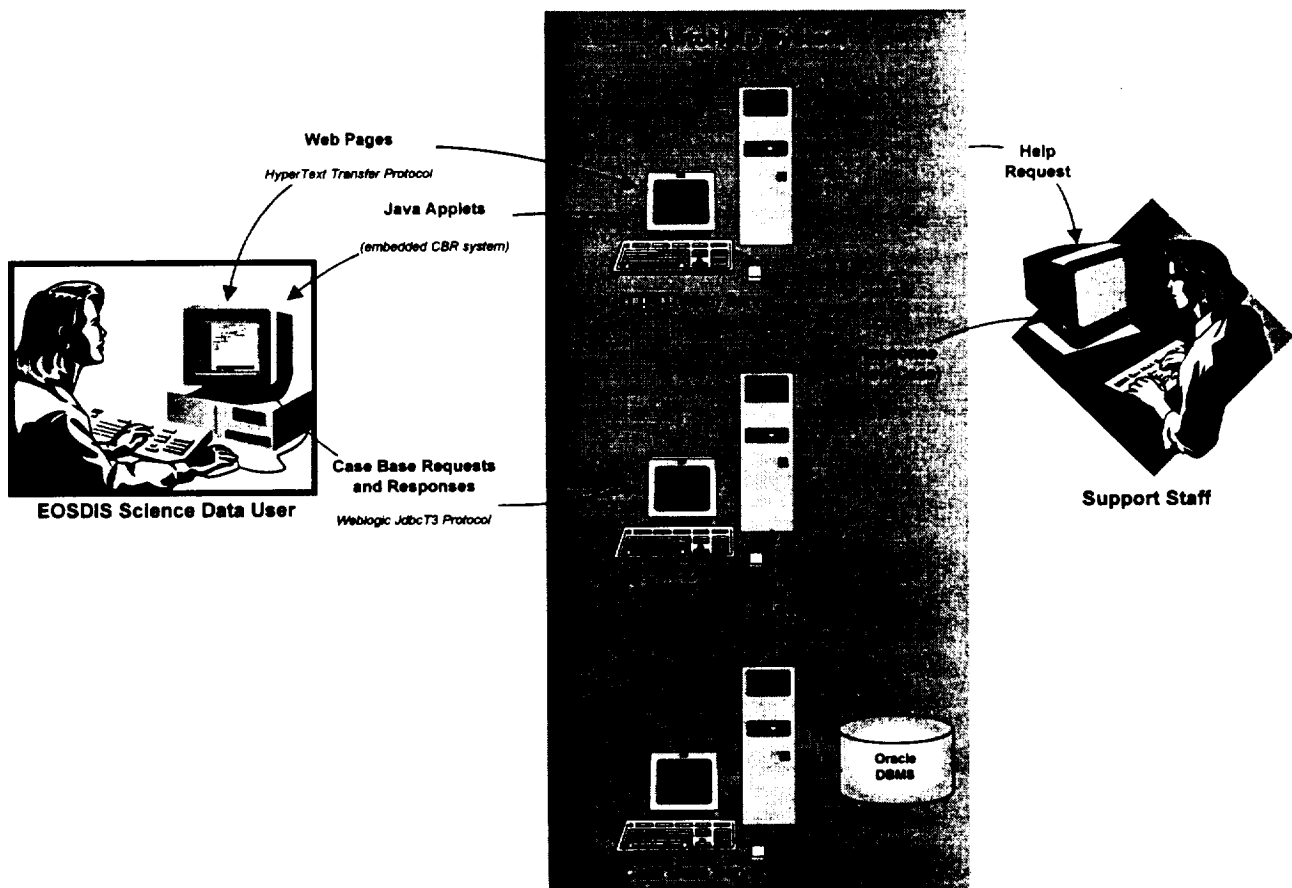


Fig. 5. AutoHelp implementation for EOSDIS

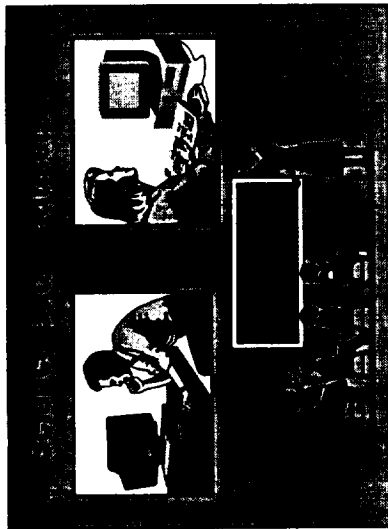
Appendix B

A Web-Accessible “Intelligent” Help Desk

PowerPoint Presentation

January 29, 1998

A Web-Accessible "Intelligent" Help Desk



An ESDIS Prototype

Christine M. Mitchell
David A. Thurman

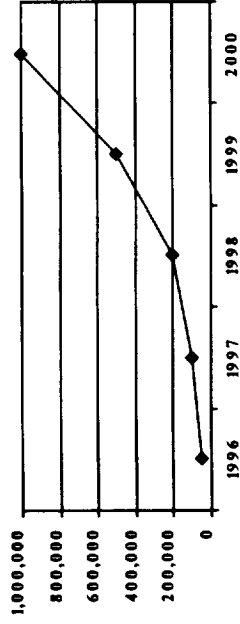
Center for Human-Machine Systems Research
School of Industrial and Systems Engineering
Georgia Institute of Technology

Walt Truszkowski, Code 522.2, Technical Monitor

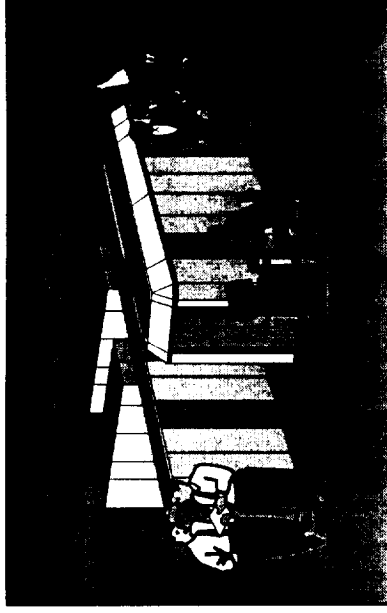
Overview of Presentation

- ❖ Introduction to year 1 goals and outputs
- ❖ Demonstration of current AutoHelp prototype
- ❖ Description of current AutoHelp architecture
- ❖ Presentation of the cognitive model of user requests
- ❖ Overview of Case-Based Reasoning to structure AutoHelp knowledge
- ❖ Current case structure
- ❖ Future Activities

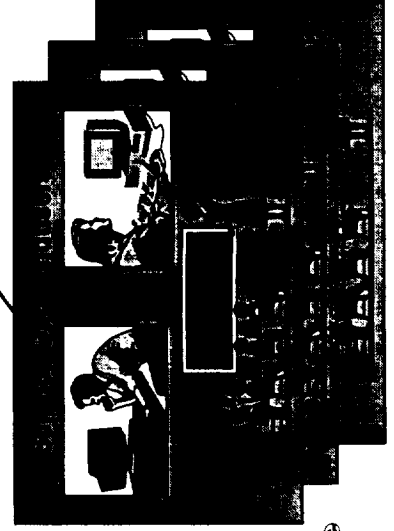
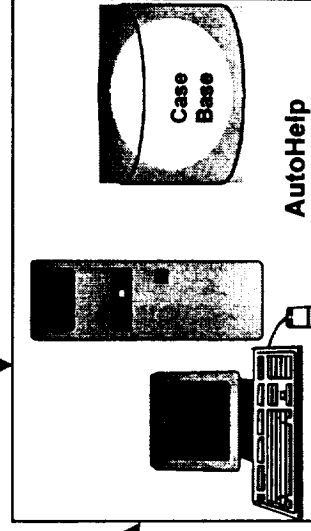
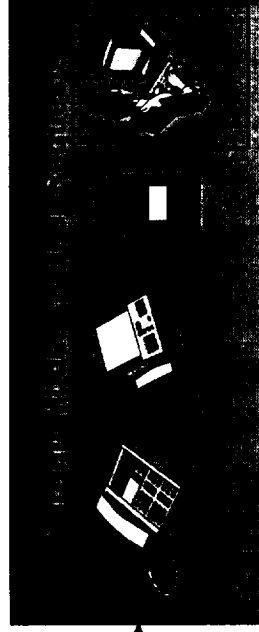
Goal: Use "technology" to meet exponential growth of EOSDIS science data users



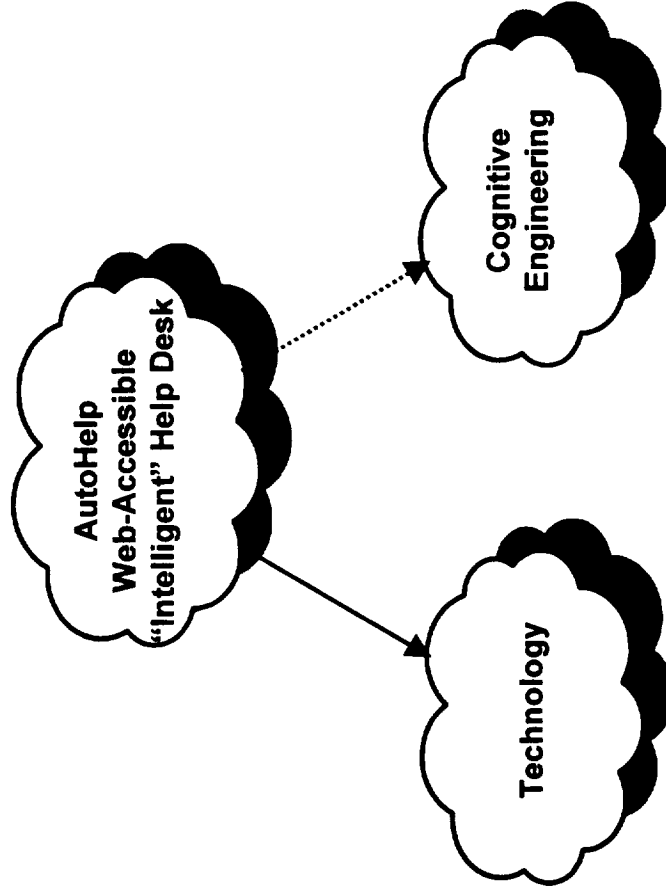
EOSDIS Enduser Population



Multinational Users



AutoHelp



- ❖ Review of COTS Help Desk Software
- ❖ Specify end-to-end help desk system
 - platform-independence
 - COTS components
 - scalable architecture
 - maintainable/extensible case base and case retrieval structure
- ❖ Field Study of Current DAAC
- ❖ Model of User Help Requests
 - multiple dimensions
 - multiple levels of abstraction
- ❖ Case-Based Reasoning
 - cases organizes existing assistance requests
 - indices provide access to previous responses



AutoHelp Screen Layout

AutoHelp

AutoHelp User Assistance Navigator

Main Categories

- Data Availability
- Ordering Data
- Data Use
- Subscriptions
- Contact Info
- Instructions
- Exit

Usage Instructions

Instructions for using the AutoHelp Navigator

AutoHelp User Assistance Navigator Instructions

AutoHelp/nav/Instructions.html

Help

nated Help Desk Project

Navigator Instructions

your problem before. If you will get a quick answer through AutoHelp. If the appropriate help desk personnel. That person will track down an answer and they will update the AutoHelp knowledge with the new information to assist you.

AutoHelp contains a Java applet which will help you navigate the knowledge base of the Goddard Space Flight Center. The Navigator will then display additional subcategories of assistance information and links to online information where appropriate.

To switch to a new main assistance category at any time, click on any of the buttons on the left side of the Navigator. To move back and forth through the series of assistance screens, use the forward and back buttons near the bottom of the Navigator. To see this instructions again, click on the Instructions button at any time.

You may notice the text "Warning: Applet Window" near the top of the AutoHelp Navigator window. This text is inserted automatically by your web browser so you know this window is from a Java applet. It is there simply to inform you and should be no cause for alarm.

Last updated: Jan 26, 1998 - 15:42:46 by duc.

Navigator Applet

- Displays case base structure
- Shows links to category information



Web Browser

- Displays assistance information based on assistance category selections



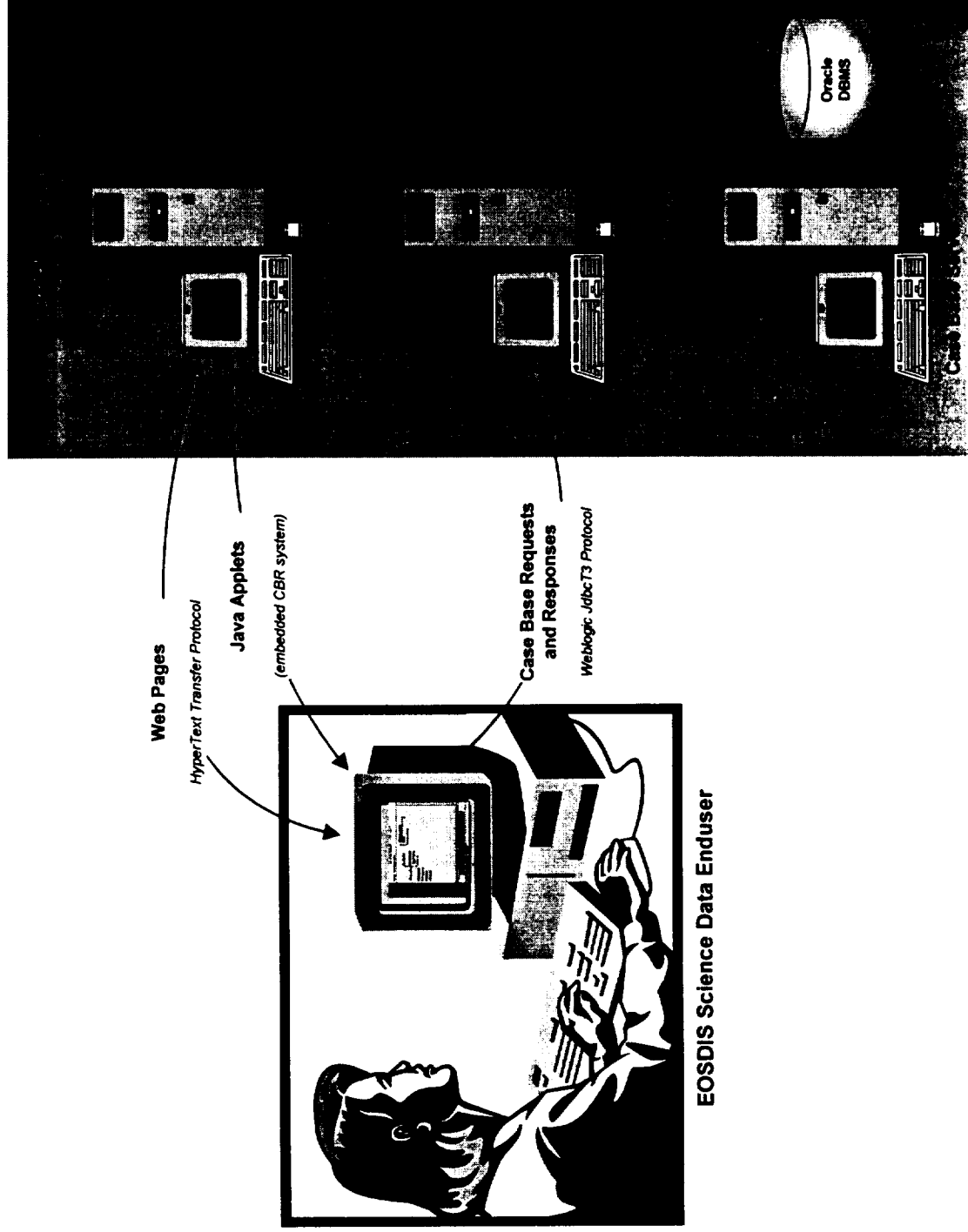
Demo: Scenario 1

- ❖ What data is available from the Goddard DAAC?
 - By scientific discipline
 - By mission/spacecraft
 - By specific dataset

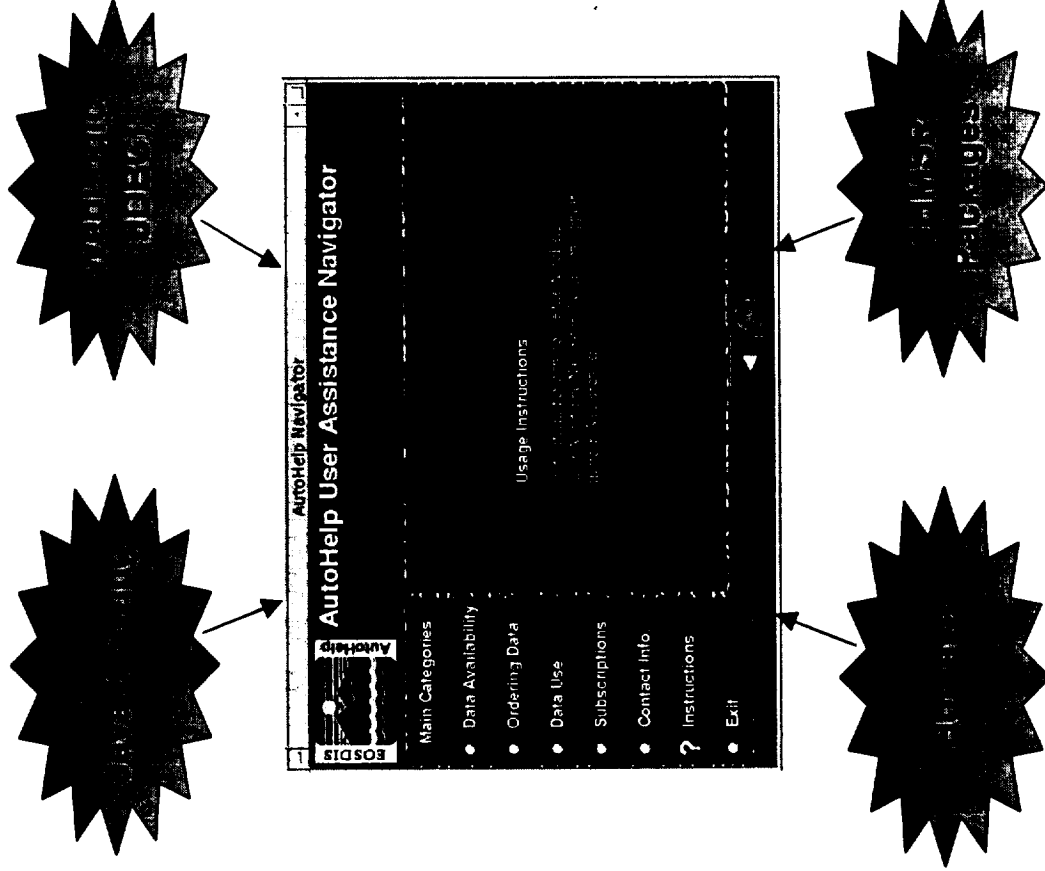
Demo: Scenario 2

- ❖ How do I order data from the Goddard DAAC?
 - What data is available on which media?
 - What's available on CD-ROM?
 - How do I order a CD-ROM?

Current AutoHelp Architecture

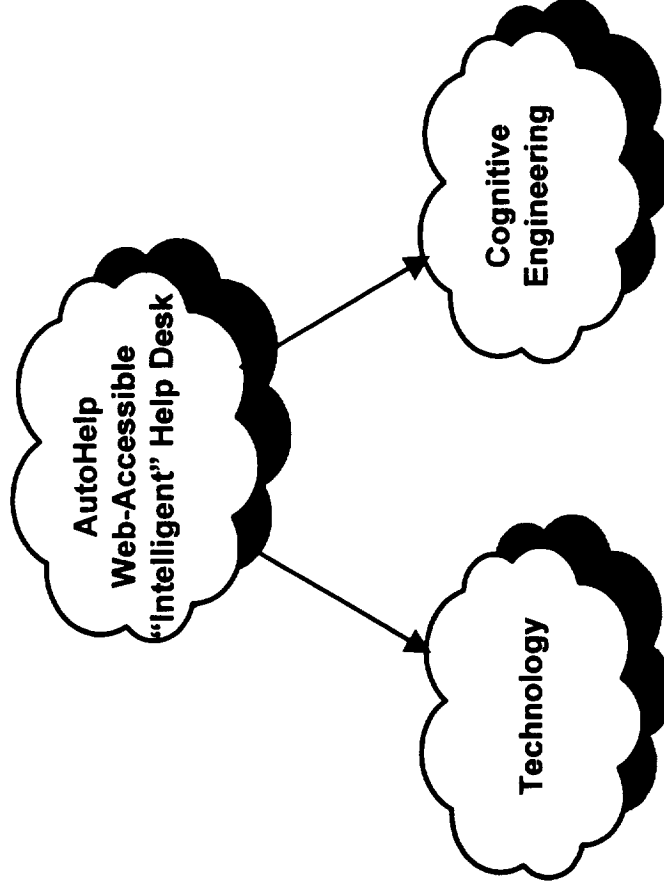


Navigator Applet Architecture



- ❖ **Java (JDK 1.1.5) & Swing (JFC)**
 - platform-independence
 - Swing provides improved performance and usability
- ❖ **WebLogic Tengah/Kona software**
 - database-independent JDBC protocol
- ❖ **HotJava**
 - Future work will include modifications to support use via Netscape and Internet Explorer
- ❖ **CHMSR Packages**
 - Homegrown software packages for easing file input/output, developing useful displays, etc.

AutoHelp

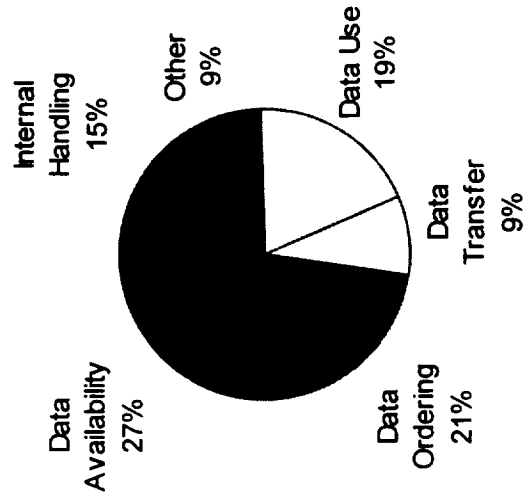


- ❖ Review of COTS Help Desk Software
- ❖ Specify end-to-end help desk system
 - platform-independence
 - COTS components
 - scalable architecture
 - maintainable/extensible case base and case retrieval structure
- ❖ Field Study of Current DAAC
- ❖ Model of User Help Requests
 - multiple dimensions
 - multiple levels of abstraction
- ❖ Case-Based Reasoning
 - cases organizes existing assistance requests
 - indices provide access to previous responses

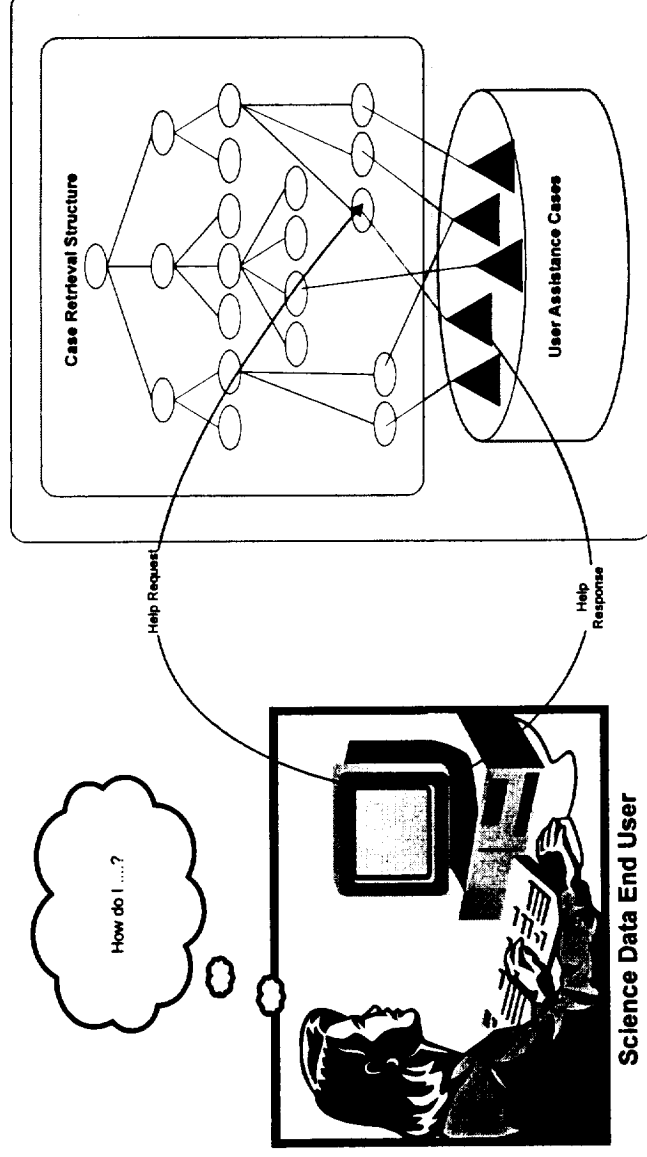
Model of Help Desk Requests

- ❖ Preliminary analysis
 - 400 email assistance requests to Goddard DAAC
- ❖ Request Categories
 - Data Availability
 - ☞ what data are available from EOSDIS?
 - Ordering Data
 - ☞ how do I order data?
 - ☞ cost?
 - Data Transfer
 - ☞ file transfer/tape reading assistance
 - Using Data
 - ☞ how do I get what I want to know from the data I have?
 - Internal Handling
 - ☞ handoffs to other DAACs

Preliminary Categorization of Assistance Requests



AutoHelp: Case-Based Reasoning



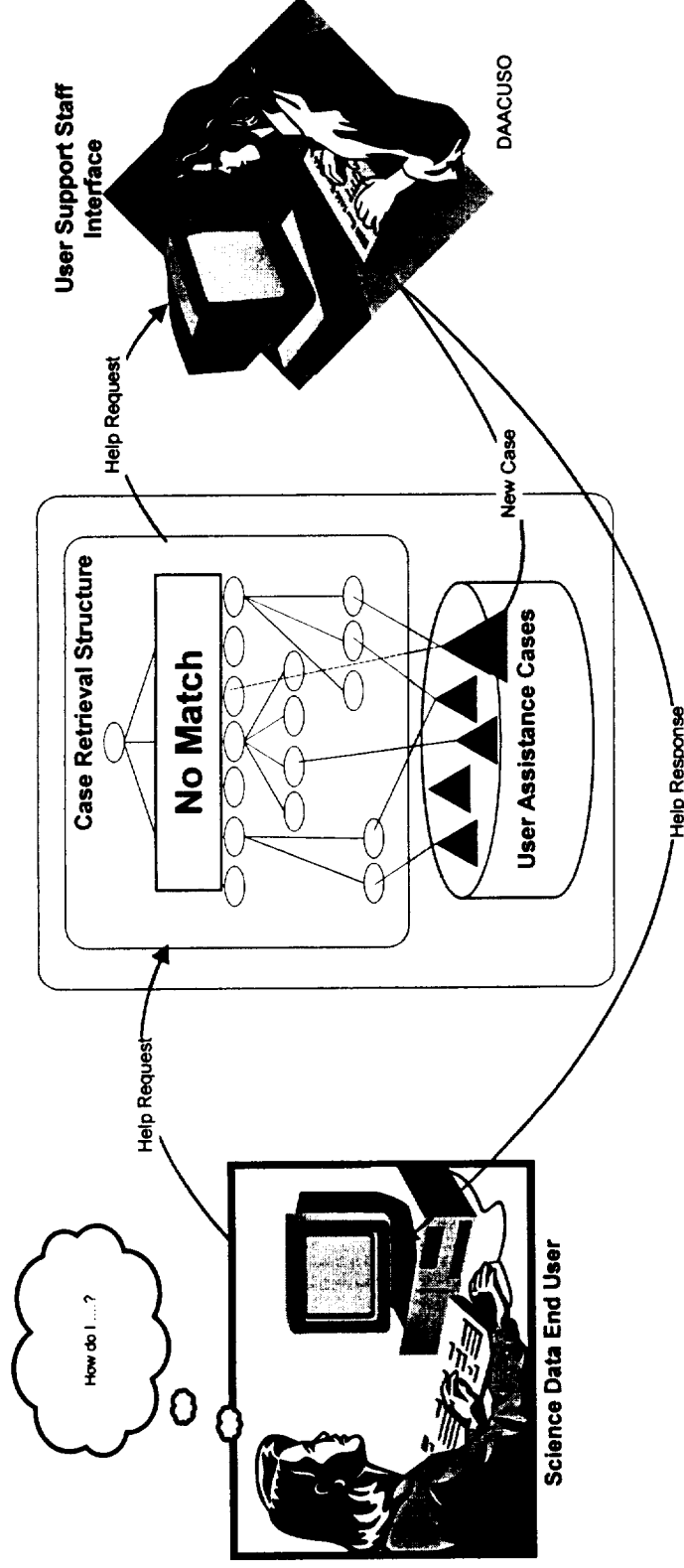
❖ Case Base

- Cases are prototypical user assistance requests and responses

❖ Case Retrieval Structure

- Case indices defined by model of user assistance requests
- Multiple dimensions (Data Availability, Ordering Data, Data Use, Software Tools, etc.)
- Multiple levels of abstraction (Data Availability: By Discipline: Ocean Color)

AutoHelp Intelligence: User Support Staff Interface



Intelligence:

- o Only answer the same questions once
- o Create new user assistance case
- o Link help request to existing user assistance cases
- o Modify case retrieval structure

Future Activities

- ❖ End-to-End Technology Demonstration
 - Refine AutoHelp Navigator applet
 - ☛ display enhancements
 - ☛ applet security
 - ☛ multiple browser support
 - Setup AutoHelp for outside access
 - ☛ install in the GSFC DAAC, or
 - ☛ maintain at Georgia Tech with DAACUSO access
 - Link illustrative user assistance requests to GSFC DAAC Web site and IMS pages
- ❖ Design & Implement DAACUSO Support Staff Interface
- ❖ Cognitive Engineering Model
 - Refine model of user assistance requests
 - ☛ questions & responses
 - Help to populate illustrative case base entries & case retrieval structure
- ❖ Usability Evaluation
 - Data collection
 - Data analysis
 - AutoHelp refinement based on evaluation

User Request with No Match

Question

I am now a member of the NASDA/RESTEC ODUS science team and also involved in ODUS Instrument design.

So I am very interested in TOMS trend data. Dr. Fumio Hasebe said you have Nimbus 7/TOMS, Meteor 3/TOMS, TOMS/UV_Erythermal Exposure, TOMS/Reflectivity version 7 CD-ROMS.

I would appreciate it very much if you could send me the above CD-ROMS to the following address. Thank you very much in advance.

Sincerely yours,
Akihiko Kuze
Electro-Optical Systems Department
Space Systems Division, NE Corp.
4035, Ikebe-cho, Tsukuki-tu, Yokohama, 224 Japan

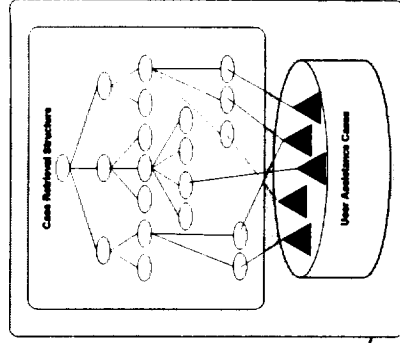
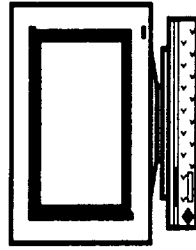
Response from DAACUSO

You can order TOMS 7 CD-ROMS from the following the Goddard DAAC Web page:
http://daac.gsfc.nasa.gov/elf-bin/elf/access_cdrom.ksh/screen=gob_entrance/test=3

User Support Staff (DAACUSO) Interface



User Support Staff Interface

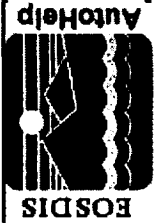


Link it into case base

Enter the new case

Question	
How can I.....	
Response	
Use your WEB Browser to request pages http://eohelp	

Case Number	<table border="1"> <tr> <td>Data Availability</td> <td>Earth System</td> </tr> <tr> <td>Ordering Data</td> <td>Scientific Discipline</td> </tr> <tr> <td>Data Transfer</td> <td>Specacraft</td> </tr> <tr> <td>Using Data</td> <td>Program/Project</td> </tr> <tr> <td>Enter New Category Type</td> <td>Specific Dataset</td> </tr> <tr> <td></td> <td>Enter New Subcategory Type</td> </tr> </table>	Data Availability	Earth System	Ordering Data	Scientific Discipline	Data Transfer	Specacraft	Using Data	Program/Project	Enter New Category Type	Specific Dataset		Enter New Subcategory Type
Data Availability	Earth System												
Ordering Data	Scientific Discipline												
Data Transfer	Specacraft												
Using Data	Program/Project												
Enter New Category Type	Specific Dataset												
	Enter New Subcategory Type												



Support Staff Interface Mockup



AutoHelp Support Staff Interface

User Question

I'm having difficulty reading the PAL data tapes I ordered. Do you have any information about how I read them on my Sun workstation with an Exabyte drive?

Thanks,
End User

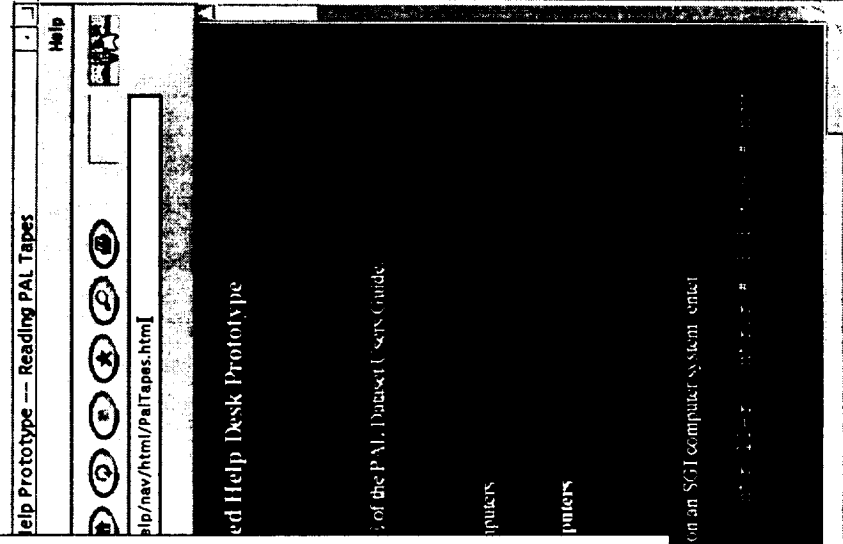
AutoHelp WebLink Response

<http://eoshelp.isye.gatech.edu/autoHelp/nav/html/PalTapes.html>

Inform User

Update AutoHelp

Update AutoHelp
and
Inform User



Informing the User



AutoHelp Support Staff Interface

To:

End User

From:

DAACUSO

Subject:

Re: Reading PAL Tapes

Message Preview:

In response to your question:

I'm having difficulty reading the PAL data tapes I ordered. Do you have any information about how I read them on my Sun workstation with an Exabyte drive?

The information you need is now available via the Web at the following location:

<http://eoshelp.isye.gatech.edu/nav/html/PalTapes.html>

If your mailer supports HTML, the information is appended below.

Sincerely,
Goddard DAAC User Support Office

Send Message

Cancel



Updating AutoHelp



AutoHelp Support Staff Interface

User Question

I'm having difficulty reading the PAL data tapes I ordered. Do you have any information about how I read them on my Sun workstation with an Exabyte drive?

Thanks,
End User

AutoHelp WebLink Response

Invoke HTML Editor

<http://eoshelp.isye.gatech.edu/autoHelp/nav/html/PalTapes.html>

Category Selection

Data Availability	Earth System	CZCS	Dataset Overview
Ordering Data	Discipline	SeaWiFS	How to order...
Data Transfer	Spacecraft	NDVI	Visualization Tools
Using Data	Program	New...	
	Dataset		
	New...		

Insert Link

Cancel

